BRIEF DESCRIPTION OF THE DRAWINGS

In order to describe the manner in which the various features can be obtained, a more particular description of the subject matter of the invention will be rendered by reference to specific embodiments which are illustrated in the appended drawings. Understanding that these drawings depict only typical embodiments and are not therefore to be considered to be limiting in scope, embodiments will be described and explained with additional specificity and detail through the use of the accompanying drawings in which:

Figures A1–A3 are diagrams representing examples of select objects in a branching algorithm;

Figures B1–B3 are diagrams representing other implementations of select objects in a branching algorithm;

Figure C1 is an illustrative drawing of an example of another implementation of a select object in a branching algorithm;

Figure C2 is a diagram illustrating a library object referenced by **Figure C1**;

Figure C3 is an illustrative drawing showing connections in a branching algorithm associated with select object of **Figure C1;**

Figure D1 is an illustrative drawing showing a top-level polymorphic ADD object that can be included within a design diagram;

Figure D2 is an illustrative drawing showing the polymorphic ADD object of **Figure D1** connected with transports;

Figures D3–D8 are illustrative drawings that show the polymorphic ADD behavior object at various stages of a synthesis process;

Figure D9 is a diagram illustrating a next level parallel add object inserted into the design in place of the top level parallel add object;

Figure D10 provides an illustration of a next level serial add object;

Figure E1A is an illustrative drawing showing a representation of a top-level cast behavior object;

Figure E1B is an illustrative drawing showing a portion of an object library that pertains to cast behavior;

Figure E1 is an illustrative drawing showing details of a variant cast object;

Figure E2 is an illustrative drawing showing a variant castup object referenced using the castup object of **Figure E1**;

Figure E3 is an illustrative drawing showing a terminal leaf castup object;

Figure E4 is an illustrative drawing showing a variant castdown object referenced using the top-level castdown object of **Figure E1**;

Figure E5 is an illustrative diagram that pictorially represents in general terms the evolution of the castup behavior object during a synthesis process;

Figure E6 is an illustrative drawing showing an explicit castdown object;

Figure E7 is an illustrative drawing showing a representation of the portion of an object library relating to castdown behavior;

Figure E8 is an illustrative drawing showing a collection of objects;

Figure E9 is an illustrative drawing showing a portion of the object library that pertains to the logical function;

Figure F1 is an illustrative drawing of a portion of an object library pertaining to an invert equivalent function;

Figure F2 is an illustrative diagram showing a representation of a top-level invert function;

Figure F3 is an illustrative diagram representing the variant invert object;

Figure F4 is an illustrative diagram showing an atomic invert object;

Figures F5–F8 are illustrative diagrams showing an evolution of an invert object in a design in response to propagation of a nibble data set to its input node;

Figure G1 is an illustrative drawing that represents several types of data set exposers and corresponding data set collectors;

Figure G2 is an illustrative drawing showing a portion of an object library related to a variant type data set exposer object;

Figure G3 is an illustrative drawing showing a portion of an object library related to a variant type data set collector object;

Figures H1–H8 provide pictorial representations of the operators of the Axiom system;

Figures I1–I5 provide pictorial representations of some special basic operators of the language;

Figures J1–J49 are illustrative diagrams showing topology of a program constructed in accordance with various embodiments of the invention;

Figure K1 is a schematic diagram of an inverter;

Figure K2 is a schematic diagram showing an inverter connected to a state element;

Figure K3 is a drawing showing a behavioral object that includes an inverter object;

Figures L1–L2 are diagrams illustrating atomic object resolution rules for an atomic INVERTER operator;

Figures L3–L4 are diagrams illustrating atomic object resolution rules for an atomic AND operator;

Figures L5–L6 are diagrams illustrating atomic object resolution rules for an atomic OR operator;

Figures L7–L10 are diagrams illustrating atomic object resolution rules for an Assignment operator;

Figures L10–L14 are illustrative drawings showing combined operation of the propagate data sets process and the flatten process in reducing a variant polymorphic behavior object into its atomic elements;

Figures M1–M3 are illustrative drawings showing a flatten process that substitutes a lower-level object into a design in place of a given composite object;

Figure N1 is a diagram of a source node;

Figure N2 is a drawing of a process that constructs the hierarchy of collector objects;

Figures O1–O3 are drawings that illustrate the operation of the remove collector and exposer objects process;

Figures P1–P7 shall be used to provide a hypothetical example of such a computational polymorphic object;

Figures P1–P2 are illustrative drawings showing examples of a one bit serial add and a two bit serial add;

Figure P3 is an illustrative drawing showing a hypothetical computational polymorphic add object;

Figure P4 is a drawing that illustrates the topology of a variant type data set shift-by-type object;

Figure P5 is a drawing that shows a topology of a terminal leaf of the variant shift-by-type object;

Figure P6 is a drawing that illustrates the topology of a variant type data set RCR object; and

Figure P7 is a drawing that illustrates the topology of a terminal leaf RCR object.

Detailed Description OF THE PREFERRED EMBODIMENT